UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/618,035	06/04/2004	Arvind Prabhakar	03226/493002; SUN040039	4145

32615 7590 02/23/2010

OSHA LIANG LLP/Oracle
TWO HOUSTON CENTER
909 FANNIN, SUITE 3500
HOUSTON, TX 77010

EXAMINER	
MAI, KEVIN S	

ART UNIT	PAPER NUMBER
2456	

NOTIFICATION DATE	DELIVERY MODE
02/23/2010	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

docketing@oshaliang.com
lord@oshaliang.com
hathaway@oshaliang.com

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/618,035
Filing Date: June 04, 2004
Appellant(s): PRABHAKAR ET AL.

Robert P. Lord
Registration No. 46479

For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed November 23, 2009 appealing from the Office action mailed July 27, 2009.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

No amendment after final has been filed.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

5892919	Nielsen	4-1999
6092100	Berstis et al.	7-2000

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

1. Claims 31-33, 36-39, 42-45 and 48 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Pat. No. 5892919 to Nielsen (hereinafter "Nielsen") and further in view of US Pat. No. 6092100 to Berstis et al. (hereinafter "Berstis").

2. **As to Claim 31,** Nielsen discloses **a system for translating domain names comprising:**

a Uniform Resource Locater (URL) detection module, configured to:

receive a URL request by a user to access a destination fully qualified domain name (FQDN) (Figure 4 of Nielsen discloses a user issuing a GET command for a network address such as a URL (400), then figure 5 discloses looking up the issued URL in the spell check cache (500). As such it is seen that because the invention looks up the issued URL in its spell check cache, that it must have received the issued URL)**, and**

a URL redirection module, configured to:

receive the invalid URL request from the URL detection module (Figure 5 of Nielsen discloses processing the requested URL to see if it can find the associated correct URL (515,520). This is seen to be part of the FQDN mapping module. Since the FQDN mapping module receives the requested URL for processing it is seen that another component must have redirected the URL to the FQDN mapping module. As such it is further seen that that component must have received the invalid URL request as well)**, and**

redirect the invalid URL request to a FQDN translation module (Figure 5 of Nielsen discloses processing the requested URL to see if it can find the associated correct URL (515,520). This is seen to be part of the FQDN mapping module. Since the FQDN mapping module receives the requested URL for processing it is seen that another component must have redirected the URL to the FQDN mapping module)**; and**

the FQDN translation module, configured to:

translate the invalid URL request to a target valid FQDN using a FQDN mapping module (Figure 5 of Nielsen discloses returning the correct URL from the originally invalid URL and then issuing that URL instead of the original URL (545, 550). Thus it is seen that the invalid URL has been translated to the correct URL)**, wherein the FQDN mapping module is stored on a computer readable storage medium** (Column 5 lines 10-20 of Nielsen disclose memory media will contain the program information for controlling the computer to enable the computer to perform its functions in accordance with the invention)**.**

Nielsen does not explicitly disclose **determine that the URL request is an invalid URL request when the URL request is inconsistent with a predefined URL stored in a cookie, wherein the predefined URL stored in the cookie is specified by the user;**

However, Berstis discloses this (Column 5 line 50 – Column 6 line 16 and figure 4 of Berstis disclose at step 52 a test is done to determine whether the string entered in the address field (URL) is recognized. An address is considered recognized if the client has made contact with that URL and thus the list of recognized URLs consists of any past URL the user has been able to access. If the string is recognized (valid) then the client is connected to the target URL, if the string is not recognized (inconsistent with predefined URL) then the process continues to try

to correct the string. It is seen that the list of recognized URLs were specified by the user because it is a list of previously contacted URLs, and in order for the URLs to be previously contacted they must have originally been specified by the user in an attempt to access those URLs)

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the URL correction as disclosed by Nielsen, with detecting invalid URLs using a list of previously contacted URLs as disclosed by Berstis. One of ordinary skill in the art would have been motivated to combine to use a known technique to improve similar devices in the same way. Nielsen and Berstis are both directed toward identifying incorrect URLs and correcting them automatically for the user. As such it would be obvious to implement the features of either invention with each other to improve similar systems in the same way.

3. **As to Claim 32,** Nielsen-Berstis discloses the invention as claimed as described in claim 31**, further comprising:**

a FQDN default setter configured to provide a predefined default target valid FQDN, wherein the FQDN default setter is used by the FQDN mapping module (Figure 5 of Nielsen discloses if the invention is unable to conclusively correct the invalid URL it will return a page to the user with the candidate URL and a request for other candidates. This is seen to be a default target valid FQDN, as it is the default if the correction to the invalid URL is not readily available. This page is seen to be predefined because it has been set by the invention to be the default page).

4. **As to Claim 33,** Nielsen-Berstis discloses the invention as claimed as described in claim 31, **wherein the FQDN mapping module is configured to provide a mapping between the invalid URL request and the target valid FQDN** (Figure 3 of Nielsen discloses a table that holds the invalid URLs and the correct URLs that they have been mapped to and then figure 5 discloses returning the correct URL from the originally invalid URL (545). This is seen to be having provided a mapping between the invalid URL and target valid FQDN).

5. **As to Claim 36,** Nielsen-Berstis discloses the invention as claimed as described in claim 31, **wherein the URL detection module, the URL redirection module, and the FQDN translation module execute in a browser** (Column 5 lines 20-25 of Nielsen disclose the user's computing device running a network browser such as a WWW browser software. Then column 2 lines 55-60 disclose the spell checking will transparently correct the URL and instruct the browser to return the document addressed by the corrected URL. Since the spell checker is able to instruct the browser it is seen to be executing inside the browser. As such it is seen that all associated modules are executing within the browser).

6. **As to Claim 37,** Nielsen discloses **a method for translating domain names, comprising:**

receiving, by a Uniform Resource Locater (URL) detection module, a URL request from a user to access a destination fully qualified domain name (FQDN) (Figure 4 of Nielsen discloses a user issuing a GET command for a network address such as a URL (400), then figure 5 discloses looking up the issued URL in the spell check cache (500). As such it is seen that

because the invention looks up the issued URL in its spell check cache, that it must have received the issued URL)**, and**

receiving, by a URL redirection module, the invalid URL request from the URL detection module (Figure 5 of Nielsen discloses processing the requested URL to see if it can find the associated correct URL (515,520). This is seen to be part of the FQDN mapping module. Since the FQDN mapping module receives the requested URL for processing it is seen that another component must have redirected the URL to the FQDN mapping module. As such it is further seen that that component must have received the invalid URL request as well)**;**

redirecting, by the URL redirection module, the invalid URL request to a FQDN translation module (Figure 5 of Nielsen discloses processing the requested URL to see if it can find the associated correct URL (515,520). This is seen to be part of the FQDN mapping module. Since the FQDN mapping module receives the requested URL for processing it is seen that another component must have redirected the URL to the FQDN mapping module)**;**

translating, by the FQDN translation module, the invalid URL request to a target valid FQDN using a FQDN mapping module (Figure 5 of Nielsen discloses returning the correct URL from the originally invalid URL and then issuing that URL instead of the original URL (545, 550). Thus it is seen that the invalid URL has been translated to the correct URL)**; and**

directing the user to a web site associated with the target valid FQDN (Figure 5 of Nielsen discloses returning the correct URL from the originally invalid URL and then issuing that URL instead of the original URL (545, 550). Thus it is seen that the invalid URL has been translated to the correct URL, which was then issued)**.**

Nielsen does not explicitly disclose **determining, by the URL detection module that the URL request is an invalid URL request when the URL request is inconsistent with a predefined URL stored in a cookie, wherein the predefined URL stored in the cookie is specified by the user**

However, Berstis discloses this (Column 5 line 50 – Column 6 line 16 and figure 4 of Berstis disclose at step 52 a test is done to determine whether the string entered in the address field (URL) is recognized. An address is considered recognized if the client has made contact with that URL and thus the list of recognized URLs consists of any past URL the user has been able to access. If the string is recognized (valid) then the client is connected to the target URL, if the string is not recognized (inconsistent with predefined URL) then the process continues to try to correct the string. It is seen that the list of recognized URLs were specified by the user because it is a list of previously contacted URLs, and in order for the URLs to be previously contacted they must have originally been specified by the user in an attempt to access those URLs)

Examiner recites the same rationale to combine used in claim 31.

7. **As to Claim 38,** Nielsen-Berstis discloses the invention as claimed as described in claim 37**, further comprising:**

providing a predefined default target valid FQDN by a FQDN default setter, wherein the FQDN default setter is used by the FQDN mapping module (Figure 5 of Nielsen discloses if the invention is unable to conclusively correct the invalid URL it will return a page to the user with the candidate URL and a request for other candidates. This is seen to be a default target

valid FQDN, as it is the default if the correction to the invalid URL is not readily available. This page is seen to be predefined because it has been set by the invention to be the default page).

8. **As to Claim 39,** Nielsen-Berstis discloses the invention as claimed as described in claim 37, **wherein the FQDN mapping module is configured to provide a mapping between the invalid URL request and the target valid FQDN** (Figure 3 of Nielsen discloses a table that holds the invalid URLs and the correct URLs that they have been mapped to and then figure 5 discloses returning the correct URL from the originally invalid URL (545). This is seen to be having provided a mapping between the invalid URL and target valid FQDN).

9. **As to Claim 42,** Nielsen-Berstis discloses the invention as claimed as described in claim 37, **wherein the URL detection module, the URL redirection module, and the FQDN translation module execute in a browser** (Column 5 lines 20-25 of Nielsen disclose the user's computing device running a network browser such as a WWW browser software. Then column 2 lines 55-60 disclose the spell checking will transparently correct the URL and instruct the browser to return the document addressed by the corrected URL. Since the spell checker is able to instruct the browser it is seen to be executing inside the browser. As such it is seen that all associated modules are executing within the browser).

10. **As to Claim 43,** Nielsen discloses **a computer readable medium comprising executable instructions for translating domain names by:**

receiving, by a Uniform Resource Locater (URL) detection module, a URL request from a user to access a destination fully qualified domain name (FQDN) (Figure 4 of Nielsen discloses a user issuing a GET command for a network address such as a URL (400), then figure 5 discloses looking up the issued URL in the spell check cache (500). As such it is seen that because the invention looks up the issued URL in its spell check cache, that it must have received the issued URL)**, and**

receiving, by a URL redirection module, the invalid URL request from the URL detection module (Figure 5 of Nielsen discloses processing the requested URL to see if it can find the associated correct URL (515,520). This is seen to be part of the FQDN mapping module. Since the FQDN mapping module receives the requested URL for processing it is seen that another component must have redirected the URL to the FQDN mapping module. As such it is further seen that that component must have received the invalid URL request as well**;**

redirecting, by the URL redirection module, the invalid URL request to a FQDN translation module (Figure 5 of Nielsen discloses processing the requested URL to see if it can find the associated correct URL (515,520). This is seen to be part of the FQDN mapping module. Since the FQDN mapping module receives the requested URL for processing it is seen that another component must have redirected the URL to the FQDN mapping module)**;**

translating, by the FQDN translation module, the invalid URL request to a target valid FQDN using a FQDN mapping module (Figure 5 of Nielsen discloses returning the correct URL from the originally invalid URL and then issuing that URL instead of the original URL (545, 550). Thus it is seen that the invalid URL has been translated to the correct URL)**; and**

directing the user to a web site associated with the target valid FQDN (Figure 5 of Nielsen discloses returning the correct URL from the originally invalid URL and then issuing that URL instead of the original URL (545, 550). Thus it is seen that the invalid URL has been translated to the correct URL, which was then issued).

Nielsen does not explicitly disclose **determining, by the URL detection module that the URL request is an invalid URL request when the URL request is inconsistent with a predefined URL stored in a cookie, wherein the predefined URL stored in the cookie is specified by the user**

However, Berstis discloses this (Column 5 line 50 – Column 6 line 16 and figure 4 of Berstis disclose at step 52 a test is done to determine whether the string entered in the address field (URL) is recognized. An address is considered recognized if the client has made contact with that URL and thus the list of recognized URLs consists of any past URL the user has been able to access. If the string is recognized (valid) then the client is connected to the target URL, if the string is not recognized (inconsistent with predefined URL) then the process continues to try to correct the string. It is seen that the list of recognized URLs were specified by the user because it is a list of previously contacted URLs, and in order for the URLs to be previously contacted they must have originally been specified by the user in an attempt to access those URLs)

Examiner recites the same rationale to combine used in claim 31.

11. **As to Claim 44,** Nielsen-Berstis discloses the invention as claimed as described in claim 43**, further comprising:**

providing a predefined default target valid FQDN by a FQDN default setter, wherein the FQDN default setter is used by the FQDN mapping module (Figure 5 of Nielsen discloses if the invention is unable to conclusively correct the invalid URL it will return a page to the user with the candidate URL and a request for other candidates. This is seen to be a default target valid FQDN, as it is the default if the correction to the invalid URL is not readily available. This page is seen to be predefined because it has been set by the invention to be the default page).

12. **As to Claim 45,** Nielsen-Berstis discloses the invention as claimed as described in claim 43**, wherein the FQDN mapping module is configured to provide a mapping between the invalid URL request and the target valid FQDN** (Figure 3 of Nielsen discloses a table that holds the invalid URLs and the correct URLs that they have been mapped to and then figure 5 discloses returning the correct URL from the originally invalid URL (545). This is seen to be having provided a mapping between the invalid URL and target valid FQDN).

13. **As to Claim 48,** Nielsen-Berstis discloses the invention as claimed as described in claim 43**, wherein the URL detection module, the URL redirection module, and the FQDN translation module execute in a browser** (Column 5 lines 20-25 of Nielsen disclose the user's computing device running a network browser such as a WWW browser software. Then column 2 lines 55-60 disclose the spell checking will transparently correct the URL and instruct the browser to return the document addressed by the corrected URL. Since the spell checker is able to instruct the browser it is seen to be executing inside the browser. As such it is seen that all associated modules are executing within the browser).

14. Claims 34, 35, 40, 41, 46 and 47 are rejected under 35 U.S.C. 103(a) as being unpatentable over Nielsen-Berstis and further in view of US Pat. 6151624 to Teare et al. (hereinafter "Teare").

15. **As to Claim 34,** Nielsen-Berstis discloses the invention as claimed as described in claim 31. Nielsen-Berstis does not explicitly disclose **wherein the URL request comprises an alias, wherein the alias is stored in the FQDN mapping module.**

However, Teare discloses this (Figure 6 of Teare discloses receiving a real name entry in a browser's network address field (602) and then looking up the real name in an override table (606). The override table is shown in figure 10 to map addresses to specific URLs)

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the system of claim 31 as disclosed by Nielsen-Berstis, with having the URL request comprise an alias and having the alias be stored in the mapping module disclosed by Teare. One of ordinary skill in the art would have been motivated to combine because it is desirable to have a way to access information available over the Web using a natural language word or "real" name associated with the information (column 4 lines 4-6 of Teare).

16. **As to Claim 35,** Nielsen-Berstis-Teare discloses the invention as claimed as described in claim 34, **wherein the FQDN mapping module comprises a mapping of the alias to the target valid FQDN** (Figure 6 of Teare discloses receiving a real name entry in a browser's

network address field (602) and then looking up the real name in an override table (606). The override table is shown in figure 10 to map addresses to specific URLs).

Examiner recites the same rationale to combine used in claim 34.

17. **As to Claim 40,** Nielsen-Berstis discloses the invention as claimed as described in claim 37. Nielsen-Berstis does not explicitly disclose **wherein the URL request comprises an alias, wherein the alias is stored in the FQDN mapping module.**

However, Teare discloses this (Figure 6 of Teare discloses receiving a real name entry in a browser's network address field (602) and then looking up the real name in an override table (606). The override table is shown in figure 10 to map addresses to specific URLs)

Examiner recites the same rationale to combine used in claim 34.

18. **As to Claim 41,** Nielsen-Berstis-Teare discloses the invention as claimed as described in claim 40**, wherein the FQDN mapping module comprises a mapping of the alias to the target valid FQDN** (Figure 6 of Teare discloses receiving a real name entry in a browser's network address field (602) and then looking up the real name in an override table (606). The override table is shown in figure 10 to map addresses to specific URLs).

Examiner recites the same rationale to combine used in claim 34.

19. **As to Claim 46,** Nielsen-Berstis discloses the invention as claimed as described in claim 43. Nielsen-Berstis does not explicitly disclose **wherein the URL request comprises an alias, wherein the alias is stored in the FQDN mapping module.**

However, Teare discloses this (Figure 6 of Teare discloses receiving a real name entry in a browser's network address field (602) and then looking up the real name in an override table (606). The override table is shown in figure 10 to map addresses to specific URLs)

Examiner recites the same rationale to combine used in claim 34.

20. **As to Claim 47,** Nielsen-Berstis-Teare discloses the invention as claimed as described in claim 46, **wherein the FQDN mapping module comprises a mapping of the alias URL request to the target valid FQDN** (Figure 6 of Teare discloses receiving a real name entry in a browser's network address field (602) and then looking up the real name in an override table (606). The override table is shown in figure 10 to map addresses to specific URLs)**.**

Examiner recites the same rationale to combine used in claim 34.

(10) Response to Argument

The examiner summarizes the various points raised by the appellant and addresses replies individually.

As per appellant's argument that:

(1) Regarding the rejection of claims 31-33, 36-39, 42-45 and 48 under 35 U.S.C. 103(a) as being unpatentable over US Pat. No. 5892919 to Nielsen (hereinafter "Nielsen") in view of US Pat. No. 6092100 to Berstis et al. (hereinafter "Berstis"), appellant argues that they disagree with the Examiner's contentions as to the teachings of Berstis. Specifically, the Examiner has ignored explicitly limitations required by the claims, and is thus failing to consider "all words in a claim". Claim 31 explicitly requires a predefined URL stored in a cookie. However, the cited

portion of Berstis is silent with regard to a predefined URL stored in a cookie. In fact, a review of Berstis reveals that Berstis is entirely silent with regard to storing any information in cookies. Further, a review of Nielsen reveals that Nielsen is also silent with regard to a predefined URL stored in a cookie. Therefore, Nielsen and Berstis, either alone or in combination, fail to disclose at least the aforementioned limitations. (Page 10 lines 12 - 20 of appellants appeal brief)

In response to argument (1), examiner asserts that applicant's usage of the term cookies is inconsistent with its common definition as understood in the art. Throughout appellants specification there are references to "user defined cookies" (Page 6 line 17 of appellants specification) or "user configured cookies" (page 8 line 8 of appellants specification), indicating that the cookies are configured by a user. However, Microsoft Computer Dictionary Fifth Edition defines cookies as: "1. a block of data that a server returns to a client in response to a request from the client. 2. on the World Wide Web, a block of data that a Web server stores on a client system. When a user returns to the same Web site, the browser sends a copy of the cookie back to the server. Cookies are used to identify users, to instruct the server to send a customized version of the requested Web page, to submit account information for the user, and for other administrative purposes." Such a definition indicates that cookies are not data that are handled by a user. Thus for this reason the term "cookies" in appellants specification was interpreted to indicate a data structure of sorts that is configured by the user as opposed to traditionally understood cookies. As such, Berstis is seen to disclose such a structure, "a given client machine includes a fuzzy URL detection engine that tests the URL against a "local" history list" (column 2 lines 28-31 of Berstis). The history list is seen to be the structure containing URLs

and since the URLs in a history list are a group of URLs that were specified by the user in an attempt to access them, it is seen that the URLs are predefined by the user.

(2) Regarding the rejection of claims 31-33, 36-39, 42-45 and 48 under 35 U.S.C. 103(a) as being unpatentable over Nielsen in view of Berstis, appellant argues that even assuming that Berstis discloses a predefined URL stored in a cookie, Berstis fails to disclose determining that the URL request is invalid if the URL request does not match the predefined URL stored in the cookie. The examiner contends that determining that the string is not recognized discloses determining that the URL request is invalid. However, Berstis is entirely silent with regard to how the test at step 52 is performed. Thus, because Berstis fails to explain how to determine how to determine that the string is not recognized, Berstis cannot possibly disclose determining that the URL request is invalid when the URL request does not match a predefined URL stored in a cookie. (Page 10 line 21 - page 11 line 10 of appellants appeal brief)

In response to argument (2), examiner asserts that Berstis' disclosure of determining the string is not recognized discloses determining that the URL request is invalid. Column 5 line 50 – Column 6 line 16 and figure 4 of Berstis disclose at step 52 a test is done to determine whether the string entered in the address field (URL) is recognized. If a URL is not recognized a test is performed to see if the string "matches" against any entry in a lexicon consisting of server IP names that have been used by the Web client over a given "history" period with respect to a given confidence level. Thus since close matches are found by comparing the URL to a history list it is seen that recognizing a string would be done in a similar manner. Namely that it would be checked to see if the string matches a string of a previously visited URL. Thus disclosure of

recognizing a URL or not recognizing a URL is seen to disclose identifying if a URL request matches or does not match predefined URLs. This is seen be to be reasonable interpretation of the recognition step since column 2 lines 60-65 of Berstis disclose checking if the URL is not fully recognized at the client as opposed to utilizing any external sources.

Furthermore it would have been obvious in view of the background of Berstis. Column 1 lines 35-45 of Berstis disclose a "look ahead" system that compares a currently typed URL with a URL list consisting of URLs that have been previously accessed from the browser during a given time period. Accordingly it would be obvious for the initial recognition to be based on comparing the typed URL with a URL list consisting of URLs that have been previously accessed from the browser during a given time period, since Berstis discloses it as a known method in the art.

(3) Regarding the rejection of claims 31-33, 36-39, 42-45 and 48 under 35 U.S.C. 103(a) as being unpatentable over Nielsen in view of Berstis, appellant argues that a review of Nielsen confirms that Nielsen fails to disclose the aforementioned requirement. Instead, Nielsen discloses determining whether an issued URL is found in a spell check cache, and if so, then determining that the issued URL is incorrect. In other words, Nielsen teaches storing incorrect URLs in the spell check cache, and determining that an issued URL is incorrect when it matches the spell check cache. In contrast, the present claims require an entirely different approach, namely storing correct URLs in cookies, and determining that a URL request is incorrect when it does not match the correct URLS stored in the cookies. (Page 11 lines 11 - 20 of appellants appeal brief)

In response to argument (3), examiner asserts that while Nielsen was not relied upon to disclose this feature at least such a feature would likely be obvious in view of Nielsen. As summarized by the appellant, Nielsen discloses a system storing incorrect URLs and identifying issued URLs as incorrect when a match occurs. While appellants system store correct URLs to identify incorrect URLs when a match does not occur. However, to the examiner, such a limitation would be somewhat obvious in view of Nielsen. Identifying that a URL is incorrect either by matching it with a list of incorrect URLs or being unable to match it with a list of correct URLs appear to be fairly obvious variants of each other. To the examiner, it would be obvious to try one variant of identifying incorrect URLs in view of the other.

(4) Regarding the rejection of claims 31-33, 36-39, 42-45 and 48 under 35 U.S.C. 103(a) as being unpatentable over Nielsen in view of Berstis, appellant argues that even assuming that Berstis discloses requirement (i), appellants submit that Berstis fails to disclose that the predefined URL stored in the cookie is specified by the user. Berstis is entirely silent regarding how to determine whether the URL string is recognized at step 52. Thus, contrary to the Examiner's contentions, Berstis does not disclose any list of URLs used at step 52 to determine whether the string is recognized. Rather, it appears that the list referred to by the examiner is the "lexicon of server IP names" described in Berstis. (Page 11 line 21 - page 12 line 9 of appellants appeal brief)

In response to argument (4), examiner asserts that Berstis discloses this feature. As argued in response to argument (2): Berstis' disclosure of determining the string is not recognized discloses determining that the URL request is invalid. Column 5 line 50 – Column 6

line 16 and figure 4 of Berstis disclose at step 52 a test is done to determine whether the string entered in the address field (URL) is recognized. If a URL is not recognized a test is performed to see if the string "matches" against any entry in a lexicon consisting of server IP names that have been used by the Web client over a given "history" period with respect to a given confidence level. Thus since close matches are found by comparing the URL to a history list it is seen that recognizing a string would be done in a similar manner. Namely that it would be checked to see if the string matches a string of a previously visited URL. Thus disclosure of recognizing a URL or not recognizing a URL is seen to disclose identifying if a URL request matches or does not match predefined URLs. This is seen be to be reasonable interpretation of the recognition step since column 2 lines 60-65 of Berstis disclose checking if the URL is not fully recognized at the client as opposed to utilizing any external sources.

Furthermore it would have been obvious in view of the background of Berstis. Column 1 lines 35-45 of Berstis disclose a "look ahead" system that compares a currently typed URL with a URL list consisting of URLs that have been previously accessed from the browser during a given time period. Accordingly it would be obvious for the initial recognition to be based on comparing the typed URL with a URL list consisting of URLs that have been previously accessed from the browser during a given time period, since Berstis discloses it as a known method in the art.

Thus, it is seen that Berstis does disclose utilizing a list of URLs to determine if a string is recognized.

(5) Regarding the rejection of claims 31-33, 36-39, 42-45 and 48 under 35 U.S.C. 103(a) as being unpatentable over Nielsen in view of Berstis, appellant argues that examiner has mischaracterized the "lexicon" of Berstis. Specifically, contrary to the examiner's assertions, Berstis fails to disclose that the "lexicon" is used to determine whether a URL string is recognized. (Page 12 line 10 - page 12 line 20 of appellants appeal brief)

In response to argument (5), examiner asserts that Berstis discloses this feature. As argued in response to argument (2): Berstis' disclosure of determining the string is not recognized discloses determining that the URL request is invalid. Column 5 line 50 – Column 6 line 16 and figure 4 of Berstis disclose at step 52 a test is done to determine whether the string entered in the address field (URL) is recognized. If a URL is not recognized a test is performed to see if the string "matches" against any entry in a lexicon consisting of server IP names that have been used by the Web client over a given "history" period with respect to a given confidence level. Thus since close matches are found by comparing the URL to a history list it is seen that recognizing a string would be done in a similar manner. Namely that it would be checked to see if the string matches a string of a previously visited URL. Thus disclosure of recognizing a URL or not recognizing a URL is seen to disclose identifying if a URL request matches or does not match predefined URLs. This is seen be to be reasonable interpretation of the recognition step since column 2 lines 60-65 of Berstis disclose checking if the URL is not fully recognized at the client as opposed to utilizing any external sources.

Furthermore it would have been obvious in view of the background of Berstis. Column 1 lines 35-45 of Berstis disclose a "look ahead" system that compares a currently typed URL with a URL list consisting of URLs that have been previously accessed from the browser during a

given time period. Accordingly it would be obvious for the initial recognition to be based on comparing the typed URL with a URL list consisting of URLs that have been previously accessed from the browser during a given time period, since Berstis discloses it as a known method in the art.

Thus, it is seen that Berstis does disclose utilizing a list of URLs to determine if a string is recognized.

(6) Regarding the rejection of claims 31-33, 36-39, 42-45 and 48 under 35 U.S.C. 103(a) as being unpatentable over Nielsen in view of Berstis, appellant argues that the URLs of the "lexicon" is not specified by the user. The examiner contends that the "lexicon" is a list of previously contacted URLs, and that "in order for the URLs to be previously contacted they must have originally been specified by the user in an attempt to access those URLs". Appellants disagree with the examiner's contentions. Appellants argue that one of skill in the art will appreciate that a web client may display a website without the URL of that website having been specified by a user. Appellants list as example, a user clicking on a hyperlinked object, automatic "popup", domain forwarding, or random selection of search results as situations where a web client may display a website without the URL of the website having been specified by a user. Therefore, a predefined URL specified by the user is not necessarily present in the "lexicon" and therefore is clearly not inherent in Berstis. (Page 12 line 22 - page 14 line 5 of appellants appeal brief)

In response to argument (6), examiner asserts that Berstis' disclosure of a lexicon of server IP names that have been used by the Web client over a given "history" period discloses

having the predefined URLs in the list being specified by the user. The server IP names in Berstis are seen to be URLs as seen in figure 7a and thus the lexicon contains URLs. Then since these URLs are URLs that have been used by the Web client over a given history period it is seen that they are predefined. Then, as argued in the office action, it is seen that the list of URLs were specified by the user because it is a list of previously contacted URLs, and in order for the URLs to be previously contacted they must have originally been specified by the user in an attempt to access those URLs.

The claim language requires the predefined URL be specified by the user. However, there is no language to explain what is meant by the term "specified". While one interpretation would include specifying a URL by typing it directly into an address bar, other interpretations would include the action of clicking on a hyperlink as specifying a URL. By clicking on a hyperlink, a user is specifying to the browser that he wants to visit the URL associated with that hyperlink. Thus it is seen that the URL added to the history was added according to being specified by the user. As such it is seen that reasonably all URLs in the history are a result of actions specified by the user and as such the URLs themselves were specified by the user.

(7) Regarding the rejection of claims 31-33, 36-39, 42-45 and 48 under 35 U.S.C. 103(a) as being unpatentable over Nielsen in view of Berstis, appellant argues that the combination of prior art references proposed by the examiner is improper. The modification proposed by the examiner would completely alter the principle of operation of Nielsen, namely the use of cached misspellings to detect spelling errors. Accordingly, by emphasizing the importance of the use of cached misspellings, Nielsen actually teaches away from the modification proposed by the

examiner. Thus, the modification proposed by the examiner is improper. (Page 14 line 7 - page 15 line 6 of appellants appeal brief)

In response to argument (7), examiner asserts that the combination is proper. As argued in response to argument (3): As summarized by the appellant, Nielsen discloses a system storing incorrect URLs and identifying issued URLs as incorrect when a match occurs. While appellants system store correct URLs to identify incorrect URLs when a match does not occur. However, to the examiner, such a limitation would be somewhat obvious in view of Nielsen. Identifying that a URL is incorrect either by matching it with a list of incorrect URLs or being unable to match it with a list of correct URLs appear to be fairly obvious variants of each other. To the examiner, it would be obvious to try one variant of identifying incorrect URLs in view of the other. Simply it would not completely alter the principle of operation of Nielsen to detect misspellings using a similar technique. With the proposed modification, Nielsen would still be able to identify URLs that are entered in incorrectly which is what Nielsen is currently performing.

(8) Regarding the rejection of claims 34, 35, 40, 41, 46 and 47 under 35 U.S.C. 103(a) as being unpatentable over Nielsen in view of Berstis in view of US Pat. 6151624 to Teare et al. (hereinafter "Teare"), appellant argues that the dependent claims are patentable over Nielsen and Berstis for at least the same reasons as their respective independent claims. Appellant further argues that Teare fails to supply that which Nielsen and Berstis lack. (Page 15 line 14 - page 16 line 13 of appellants appeal brief)

In response to argument (8), examiner asserts that the claims are disclosed by Nielsen, Berstis and Teare for the rationale set forth in the responses to arguments (1) - (7).

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Kevin S Mai/

Examiner, Art Unit 2456

Conferees:

/KEVIN BATES/
Primary Examiner, Art Unit 2456
/Rupal D. Dharia/

Supervisory Patent Examiner, Art Unit 2400